

WHAT IS CLAIMED IS:

54917 1. A program control apparatus for controlling execution of a program in a computer system in which thread is switched in accordance with scheduling by a scheduler, comprising:

5 first means responsive to a predetermined first application program interface call from a thread for setting a prescribed flag to one of first and second states;

second means for detecting, after said flag is set to said one state, a prescribed change in a state of said computer system, and for setting said flag to the other one of said first and second states; and

10 third means responsive to a predetermined second application program interface call constituting a pair with said first application program interface, from said thread, for returning a value indicative of the state of said flag to said thread.

2. The program control apparatus according to claim 1, wherein said first means includes means responsive to an application program interface call from a thread which interface requests start of detection of presence/absence of a context switching, for setting a flag indicating presence/absence of a context switching to a state indicating absence of a context switching;

5 said second means includes means for setting, after said flag is set to the state corresponding to the absence of a context switching and a scheduler switches a context, said flag to a state corresponding to presence of a context switching; and

10 said third means includes means responsive to an application program interface call from said thread which interface requests termination of detection of presence/absence of a context switching, for returning a value corresponding to the state of said flag to said thread.

815 3. The program control apparatus according to claim 2, further comprising means for invalidating, when there is said context switching,

processing of said thread from said application program interface call requiring start of detection of presence/absence of a context switching until  
5 said application program interface call requesting termination of detection of presence/absence of a context switching.

4. The program control apparatus according to claim 3, further comprising:

means for alternately changing priority of said thread to high and low; and

5 means receiving a process time from said application program interface call requesting start of detection of presence/absence of a context switching until said application program interface call requesting termination of detection of presence/absence of a context switching, for  
10 comparing the received process time with a remaining time until the priority of said thread is changed to low, when said thread is at a high priority state, and upon detection that said remaining time is shorter than said process time, lowering the priority of said thread.

5. The program control apparatus according to claim 3, wherein said thread is a garbage collection thread in accordance with copy method in which an object which is referenced by any other object in a memory heap area is detected, and the object is copied to a prescribed area in said  
5 heap area.

6. The program control apparatus according to claim 3, wherein said thread is a memory compaction thread for eliminating fragmentation, by freeing a memory area of an object not referenced by any other object in a memory heap area as a free memory area allocatable to other object.

7. A program control apparatus according to claim 1, wherein said first means includes means responsive to an application program interface call from a thread which interface request start of detection of presence/absence of a data write to a designated memory area,

5 for setting a flag indicating presence/absence of a data write to a state corresponding to absence of a data write;

said second means includes means for setting, when there is a data write to said designated memory area, said flag to a state corresponding to presence of a data write; and

10 said third means includes means responsive to an application program interface call from said thread which interface requests termination of detection of presence/absence of a data write to the designated memory area, for returning a value corresponding to the state of said flag to said thread.

8. A program control apparatus, comprising:

5 means for detecting an object in a memory heap area, which is not referenced by any object and for incrementally executing a garbage collection thread for freeing a memory area of said non-referenced object as a free memory area allocatable to other objects;

means for scheduling time divisional execution of threads in accordance with priorities of the threads; and

10 means for alternately changing priority of said garbage collection thread to be higher and lower than priorities of the threads other than the garbage collection thread.

9. The program control apparatus according to claim 8, further comprising first means for setting time in which said garbage collection thread has high priority by an application program interface call.

10. The program control apparatus according to claim 8, further comprising means for setting a period of high priority state and low priority state of said garbage collection thread by an application program interface call.

11. The program control apparatus according to claim 7, further comprising means for detecting change with time of capacity of said free

area, and

5 means for making long a time when said garbage collection thread has high priority when the capacity of said free area tends to decrease.

12. A program control apparatus including means for executing a real time thread in response to an occurrence of an event, and for executing a non-real time thread at an interruption or termination of the real time thread, wherein

5 one said non-real time thread is a garbage collection thread for detecting an object not referenced by any object in a memory heap area, and for incrementally executing garbage collection for freeing a memory area of said non-referenced object as a free memory area allocatable to other objects;

10 said program control apparatus further comprising means for executing said garbage collection thread when said free memory area in the heap area is decreased to a prescribed amount due to execution of a non-real time thread other than said garbage collection thread.

13. A program control apparatus, comprising:

5 means for detecting an object which is not referenced by any object in a memory heap area, and for selectively executing a plurality of garbage collection threads of different procedures for freeing a memory area of said non-referenced object as a free memory area allocatable to other objects; and

means for executing a garbage collection thread of one of the plurality of procedures based upon an amount of said free area or of an area used by said non-referenced object.

14. A memory allocating apparatus, comprising:

means for detecting distribution of sizes of objects allocated in a memory heap area; and

means for determining, as a size to be allocated to a new object in

5 said heap area, an integer multiple of a fixed size larger than the center of said distribution.

15. A memory allocating apparatus according to claim 14, wherein said means for detecting distribution of sizes of objects is implemented by a program module incorporated in a system when use of the system starts, and separated from the system after said distribution of sizes of objects is  
5 detected.

16. The memory allocating apparatus according to claim 14, wherein said means for determining a fixed size is implemented by a program module incorporated in a system when use of the system starts and separated from the system after said fixed size is determined.

17. A memory allocating apparatus including means for creating an object in a memory heap area, comprising:  
means for determining, as a size to be allocated to an object in said heap area, an integer multiple of a fixed size; and  
5 means for setting said fixed size in response to an application program interface call.

18. A memory allocating apparatus comprising:  
means for creating an object in a memory heap area,  
means for determining, as a size to be allocated to an object in said heap area, an integer multiple of a fixed size;  
5 means responsive to an application program interface call for setting distribution of sizes of objects allocated in said heap area; and  
means responsive to the application program interface call for setting said fixed size to a value larger than the center of said distribution.

19. A memory allocating apparatus for allocating a template class and an object created from said class in a memory heap area, comprising  
means for storing data corresponding to a time when an object is

created from said class,

5 means for detecting life of said object when said object is deleted, and providing data of said life in said class, and

means for dividing, when an object is created from said class, an area for creating the object in said heap area based on said data of life.

20. A memory allocating apparatus, comprising:

reference information storing means for detecting an object referenced by another object in a memory heap area, and for storing state of presence/absence of said reference; and

5 object deleting means for freeing a memory area of an object which is not referenced by any object as a free memory area allocatable to other object, based on the stored content of said reference information storing means; wherein

10 said reference information storing means includes reference object detecting means for storing first data of a tree structure representing a relation of reference among objects and second data representing an object at a portion where the relation of reference among objects is changed, for detecting an object which is referenced by searching the first data, for reading the second data, and for detecting an object which is referenced by  
15 searching the first data based on said read data.

21. The memory allocating apparatus according to claim 20, wherein said referenced object detecting means stores, only when an object detected for the first time by searching of said first data is referenced by said object after reference relation is changed, said referenced object as said  
5 second data.

22. A memory allocating apparatus including means for creating an object in a memory heap area, comprising:

heap area dividing means for dividing said heap area into a plurality of regions of different sizes in advance; and

5 means for allocating to an object to be created one of the divided

regions which is the smallest among those larger than the size of said object.

23. The memory allocating apparatus according to claim 22, wherein said heap area dividing means includes means for dividing said heap area by an application program interface call using number of division of said plurality of sizes and respective sizes as arguments.

5 24. A method of program control, comprising the steps of:  
in response to an application program interface call from a thread which interface requests start of detection of presence/absence of a context switching, setting a flag indicating presence/absence of a context switching to a state corresponding to absence of a context switching;

after said flag is set to the state corresponding to the absence of a context switch, when a context is switched by a scheduler, setting said flag to a state corresponding to presence of a context switching; and

10 in response to an application program interface call from said thread which interface requests termination of detection of presence/absence of a context switching, returning a value corresponding to the state of said flag to said thread.

25. A method of program control, comprising the steps of:  
in response to an application program interface call from a thread which interface requests start of detection of presence/absence of a data write to a designated memory area, setting a flag indicating  
5 presence/absence of a data write to a state corresponding to absence of the data write;

setting said flag to a state corresponding to presence of a data write when there is a data write to said designated memory area; and

10 in response to an application program interface call from said thread which interface requests termination of detection of presence/absence of a data write to the designated memory area, returning a value corresponding to the state of said flag to said thread.

26. A method of program control, comprising the steps of:  
detecting an object which is not referenced by any object in a memory  
heap area, and incrementally executing a garbage collection thread for  
freeing a memory area of said non-referenced object as a free memory area  
5 allocatable to other objects;  
scheduling time divisional execution of threads in accordance with  
priorities of the threads; and  
alternately changing priority of said garbage collection thread to be  
higher and lower than priorities of the threads other than the garbage  
10 collection thread.

27. A method of program control, comprising the step of:  
executing a real time thread in response to an occurrence of an event,  
and executing a non-real time thread at an interruption or termination of  
said real time thread; wherein  
5 one said non-real time thread is a garbage collection thread for  
detecting an object which is not referenced by any object in a memory heap  
area, and freeing a memory area of said non-referenced object as a free  
memory area allocatable to other objects;  
said method further comprising the step of  
10 executing said garbage collection thread when said free memory area  
in the heap area is decreased to a prescribed amount due to execution of a  
non-real time thread other than said garbage collection thread.

28. A method of program control, comprising the steps of  
detecting an object which is not referenced by any object in a memory  
heap area,  
freeing the memory area of said non-referenced object as a free  
5 memory area allocatable to other objects, and  
selectively executing a plurality of garbage collection threads having  
different procedures, based on an amount of area used by said object or said  
free area.



29. A method of memory allocation, comprising the steps of:  
detecting distribution of sizes of objects allocated in a memory heap  
area, and

5 determining, as a size to be allocated to a new object in said heap  
area, an integer multiple of a fixed size larger than the center of the  
distribution.

30. A method of allocating a memory for allocating a template  
class and an object created from said class in a memory heap area,  
comprising the steps of:

5 storing data corresponding to a time when an object is created from  
said class,

detecting a life of the object when said object is deleted, and  
providing data of said life in the class and

when an object is created from said class, dividing an area for  
creating an object in said heap area based on said data of life.

31. A method of memory allocation including the steps of  
detecting an object which is referenced by another object in a  
memory heap area and storing a state of presence/absence of said reference,  
and freeing a memory area of an object which is not referenced by any  
5 object as a free memory area allocatable to other objects based on the stored  
content, said method further comprising the steps of

storing first data of a tree structure representing a relation of  
reference among objects when an object is created,

10 storing second data representing an object at a portion where the  
relation of reference among objects is changed, detecting an object which is  
referenced by searching the first data, reading the second data and  
searching the first data based on said read data to detect an object which is  
referenced.

32. A computer readable recording medium storing a program  
control program allowing a computer to execute a program control method,

said program control method includes the steps of:  
 in response to an application program interface call from a thread  
 5 which interface requests start of detection of presence/absence of a context  
 switching, setting a flag indicating presence/absence of a context switching  
 to a state corresponding to absence of a context switching;

after said flag is set to the state corresponding to the absence of a  
 context switch, when a context is switched by a scheduler, setting said flag  
 10 to a state corresponding to presence of a context switching; and

in response to an application program interface call from said thread  
 which interface requests termination of detection of presence/absence of a  
 context switching, returning a value corresponding to the state of said flag  
 to said thread.

33. A computer readable recording medium storing a program  
 control program allowing a computer to execute a program control method,  
 said program control method including the steps of

in response to an application program interface call from a thread  
 which interface requests start of detection of presence/absence of a data  
 write to a designated memory area, setting a flag indicating  
 5 presence/absence of a data write to a state corresponding to absence of a  
 data write;

setting said flag to a state corresponding to presence of a data write  
 when there is a data write to said designated memory area; and

in response to an application program interface call from said thread  
 10 which interface requests termination of detection of presence/absence of a  
 data write to the designated memory area, returning a value corresponding  
 to the state of said flag to said thread.

34. A computer readable recording medium storing a program  
 control program allowing a computer to execute a program control method,  
 said program control method including the steps of

detecting an object which is not referenced by any object in a memory  
 5 heap area, and incrementally executing a garbage collection thread for

freeing a memory area of said non-referenced object as a free memory area allocatable to other objects;

scheduling time divisional execution of threads in accordance with priorities of the threads; and

10 alternately changing priority of said garbage collection thread to be higher and lower than priorities of the threads other than the garbage collection thread.

35. A computer readable recording medium storing a program control program allowing a computer to execute a program control method, said program control method including the steps of

executing a real time thread in response to an occurrence of an event, and executing a non-real time thread at an interruption or termination of said real time thread; wherein

5 one said non-real time thread is a garbage collection thread for detecting an object which is not referenced by any object in a memory heap area, and freeing a memory area of said non-referenced object as a free memory area allocatable to other objects;

10 said program control method further comprising the step of executing said garbage collection thread when said free memory area in the heap area is decreased to a prescribed amount due to execution of a non-real time thread other than said garbage collection thread.

36. A computer readable recording medium storing a program control program allowing a computer to execute a program control method, said program control method including the steps of

5 detecting an object which is not referenced by any object in a memory heap area, and freeing the memory area of said non-referenced object as a free memory area allocatable to other objects, and

selectively executing a plurality of garbage collection threads having different procedures, based on an amount of area used by said object or said free area.

37. A computer readable recording medium storing a memory allocation program allowing a computer to execute a memory allocation method creating an object in a memory heap area when a program is executed:

5        said method of memory allocation including the steps of  
      detecting distribution of sizes of objects allocated in a memory heap  
      area, and  
      determining, as a size to be allocated to a new object in said heap  
10        area, an integer multiple of a fixed size larger than the center of the  
      distribution.

38. A computer readable recording medium storing a program allowing a computer to function as a memory allocation apparatus including

5        means for creating an object in a heap area,  
      means for setting a fixed size by an application program interface  
      call, and  
      means for detecting as a size to be allocated to an object in said heap  
      area, an integer multiple of said fixed size.

39. A computer readable recording medium storing a program allowing a computer to function as a memory allocating apparatus including

5        means for creating an object in a memory heap area,  
      means for setting a distribution of sizes of object allocated in said  
      heap area by an application program interface call,  
      means responsive to said application program interface call, for  
      setting as a fixed size, a value larger than the center of said distribution,  
10        and

      means for determining, as a size to be allocated to an object in said  
      heap area, an integer multiple of said fixed size.

40. A computer readable recording medium storing a memory

allocation program allowing a computer to execute a method of memory allocation, for creating an object in a memory heap area by a class as a template when a program is executed,

5        said method of memory allocation including the steps of  
storing data corresponding to a time when an object is created from  
said class,

detecting a life of the object when said object is deleted, and  
providing data of said life in the class, and

10        when an object is created from said class, dividing an area for  
creating an object in said heap area based on said data of life.

41.    A computer readable recording medium storing a memory allocation program allowing a computer to execute a method of memory allocation by detecting an object referenced by other object in a memory heap area, storing presence/absence of the reference and deleting an object based on the stored contents,

5        said method of memory allocation including the steps of  
storing first data of a tree structure representing a relation of  
reference among objects when an object is created, and

10        storing second data representing an object at a portion where the  
relation of reference among objects is changed, detecting an object which is  
referenced by searching the first data, reading the second data, and  
searching the first data based on said read data to detect an object which is  
referenced.

42.    A computer readable recording medium storing a program allowing a computer to function as a memory allocating apparatus including

5        means for creating an object in a heap area,  
means for dividing said heap area into a plurality of sizes in advance,  
and

means for allocating to an object to be created, one of the divided  
area which is the smallest among those larger than the size of the object.